

高等职业教育计算机类专业规划教材



Java&JSP

应用程序实例开发

潘国荣 主编 殷存举 徐栋 副主编



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

高等职业教育计算机类专业规划教材

Java & JSP 应用程序实例开发

潘国荣 主 编
殷存举 徐 栋 副主编

电子工业出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书主要内容分两大部分,第一部分包括项目一至项目三,详细介绍了利用 Java 开发桌面数据库系统的方法和步骤;第二部分包括项目四至项目六,详细介绍了利用纯 JSP 页面模式、JSP+JavaBean 模式、JSP+JavaBean+Servlet 模式、Struts 模式开发 JSP 应用程序的方法和步骤。每个项目均按“项目描述”、“项目预览”、“项目分析”、“项目设计”、“环境需求”、“项目知识点”和具体任务进行组织,其中“项目实施”以任务的形式给出了完整的实施步骤和代码。

本书是作者多年从事教学和开发经验的总结,项目实施步骤详尽、注重实用,非常适合作为高等职业教育计算机类专业的教材,也可以作为开发人员和自学爱好者的参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

Java & JSP 应用程序实例开发/潘国荣主编.--北京:电子工业出版社,2014.10

ISBN 978-7-121-24111-6

I. ①J… II. ①潘… III. ①JAVA 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2014)第 188923 号

策划编辑:朱怀永

责任编辑:朱怀永

特约编辑:王 纲

印 刷:三河市鑫金马印装有限公司

装 订:三河市鑫金马印装有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:20.25 字数:518 千字

版 次:2014 年 10 月第 1 版

印 次:2014 年 10 月第 1 次印刷

印 数:3000 册

定 价:45.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换,若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zltts@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

前 言

Java 是一种可以撰写跨平台应用程序的程序设计语言,它具有面向对象、与平台无关、安全机制、高可靠性和支持网络编程的特性,自面世后就非常流行,发展迅速,对其他语言形成了有力冲击。Java 是被广泛使用的一种语言,应用于各种软件的开发中,具备显著优势和广阔前景。同时,随着网络技术的不断发展,大量软件需要采用 B/S 结构来实现,基于 Java 技术的动态网页开发技术 JSP 由于其跨平台、强大的可伸缩性、支持服务器端组件等特性而被广泛应用于 Web 应用程序的开发中。

目前,学习 Java 和 JSP 开发的人员众多、层次多样,讲述 Java 和 JSP 程序设计的教材也很多,并且各有特色。本书主要有以下几个特点:

1. 本书是作者多年教学和开发工作的总结,书中的项目都经过教学实际的检验,有些项目已经经过多年的实际应用,具有较高的学习和参考价值。
2. 项目和任务的编排符合认知规律。项目由浅入深,由简单到复杂,任务的分解安排完全按照项目开发的实际过程进行,易于上手和理解。
3. 项目实用性强。项目中的很多模块和代码段是 Java 和 JSP 程序开发中非常常用的,可以方便地在其他项目开发中加以使用。

本书按照“项目—任务”的形式编写。全书分为两大部分,第一部分包括用户登录程序、简单的列表框程序、学生成绩管理系统三个项目,详细介绍了利用 Java 开发桌面数据库系统的方法和步骤;第二部分包括网上用户注册程序、学生基本信息维护系统、教师测评系统(Struts 模式)三个项目,详细介绍了利用纯 JSP 页面模式、JSP+JavaBean 模式、JSP+JavaBean+Servlet 模式、Struts 模式开发 JSP 应用程序的方法和步骤。每个项目均按“项目描述”、“项目预览”、“项目分析”、“项目设计”、“环境需求”、“项目知识点”和具体任务进行组织,其中“项目实施”以任务的形式给出了完整的实施步骤和代码。

本书由潘国荣主编。其中潘国荣编写项目三和项目六,并对全书进行统稿,徐栋编写项目一和项目二,殷存举编写项目四和项目五,在此对相关参与教师表示诚挚的感谢。

由于编写时间仓促和水平有限,书中难免存在不足之处,恳请读者赐教斧正。

目 录

项目一 用户登录程序	1
任务一 准备数据库,定义窗体类,编制界面	2
任务二 编制事件处理代码	5
项目二 简单的列表框程序	7
任务一 定义窗体类,编制界面	7
任务二 编制事件处理代码	9
项目三 学生成绩管理系统	11
任务一 准备数据库,编制主窗体	14
任务二 编制教师基本情况维护模块	16
任务三 编制课程基本情况维护模块	26
任务四 编制班级基本情况维护模块	28
任务五 表格及表格模型	33
任务六 编制学生基本情况维护模块	40
任务七 编制班级选课模块	47
任务八 编制单科方式录入学生成绩模块	56
任务九 编制多科方式录入学生成绩模块	62
任务十 编制单个学生成绩查询模块	63
任务十一 编制班级成绩明细查询模块	67
任务十二 编制班级成绩汇总数据查询模块	71
任务十三 编制同学科成绩比较数据查询模块	75
任务十四 编制班级不及格人数统计模块	79
任务十五 编制用户登录模块并打包	83
项目四 网上用户注册程序	86
任务一 用纯 JSP 页面模式实现	88
任务二 用 JSP+JavaBean 模式实现	98
任务三 用 JSP+JavaBean+Servlet 模式实现	105
任务四 用 Struts 模式实现	110
项目五 学生基本信息维护系统	118
任务一 用 JSP+JavaBean 模式实现学生信息列表	120
任务二 用 JSP+JavaBean 模式实现学生信息查询	127
任务三 用 JSP+JavaBean 模式实现学生数据维护	130
任务四 用 JSP+JavaBean 模式实现学生数据分页显示	142



任务五	用 JSP+JavaBean+Servlet 模式实现学生信息列表	144
任务六	用 JSP+JavaBean+Servlet 模式实现学生信息查询	148
任务七	用 JSP+JavaBean+Servlet 模式实现学生数据维护	150
任务八	用 Struts 模式实现学生信息列表	154
任务九	用 Struts 模式实现学生信息查询	159
任务十	用 Struts 模式实现学生数据维护	161
项目六	教师测评系统(Struts 模式)	170
任务一	创建工程、准备数据库、编制首页面	181
任务二	编制系统管理子系统登录模块	184
任务三	编制教师基本情况维护模块	193
任务四	编制课程基本情况维护模块	210
任务五	编制班级基本情况维护模块	213
任务六	编制班级科目维护模块	232
任务七	编制用户情况维护模块	248
任务八	编制系统初始化模块	259
任务九	编制开放或禁止测评系统模块	263
任务十	编制汇总班主任测评数据模块	266
任务十一	编制汇总任课教师测评数据模块	270
任务十二	编制测评班主任模块的学生登录部分	273
任务十三	编制测评班主任模块	281
任务十四	编制测评任课教师模块的学生登录部分	289
任务十五	编制测评任课教师模块	293
任务十六	编制数据查询子系统登录模块	298
任务十七	编制查询班主任汇总数据模块	303
任务十八	编制按学部查询班主任汇总数据模块	308
任务十九	编制按班级查询任课教师汇总数据模块	310
任务二十	编制按学科查询任课教师汇总数据模块	313
任务二十一	编制按班级查询意见、建议和简要评价模块	315

项目一 用户登录程序

项目描述

用户登录程序是一个简单的 Java 桌面应用程序,它能够检查登录用户的合法性。

项目预览

运行用户登录程序时,出现如图 1-1 所示窗体。

输入用户名及口令后,窗体如图 1-2 所示。

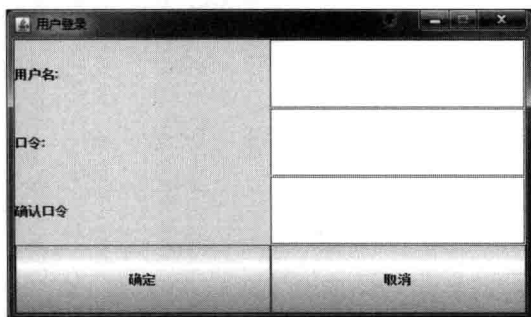


图 1-1 “用户登录”窗体

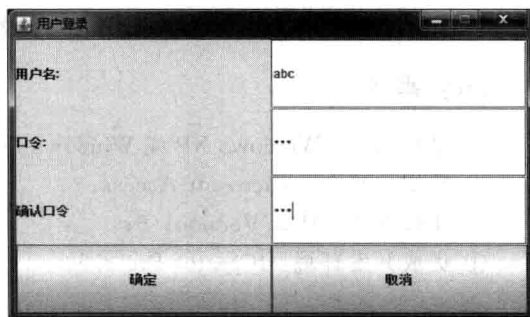


图 1-2 输入数据后的用户登录窗体

单击“确定”按钮后,在数据库表中查找是否有与该用户名、口令一致的记录,若存在,弹出“登录成功”的对话框,如图 1-3 所示。

若不存在,弹出“数据库中不存在该用户”的对话框,如图 1-4 所示。

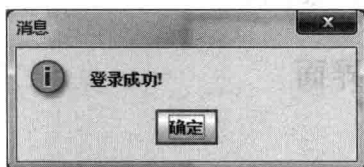


图 1-3 登录成功对话框

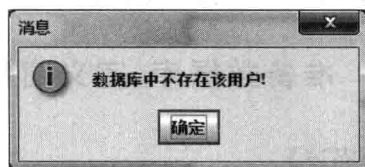


图 1-4 数据库中不存在该用户对话框

若未输入用户名,弹出一对话框,如图 1-5 所示。

若未输入口令,弹出一对话框,如图 1-6 所示。

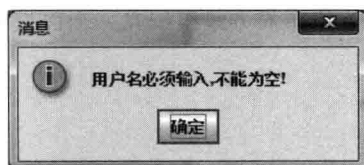


图 1-5 提示用户名必须输入的对话框

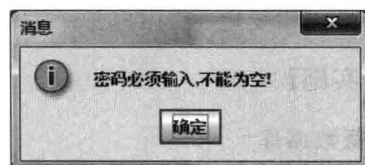


图 1-6 提示密码必须输入的对话框



若两次输入的口令不一致,弹出一对话框,如图 1-7 所示。

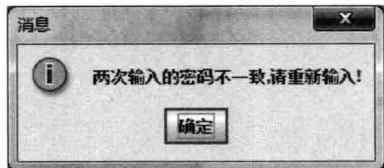


图 1-7 两次输入的密码不一致时弹出的对话框

单击“取消”按钮后退出应用程序。

项目分析

一个完整的桌面应用程序都需要有登录功能。登录时,根据用户名及密码到数据库表中去核对是否存在该用户,主要涉及对数据库表进行查询操作。

环境需求

- ① 操作系统: Windows XP 或 Windows 7;
- ② 数据库平台: Microsoft Access;
- ③ JDK 版本: JDK Version1.6;
- ④ 集成开发环境: JCreator Pro 3.5。

项目知识点

- ① Swing 中常用容器 JFrame、JPanel 的使用方法;
- ② 布局管理器及常见布局方式;
- ③ 按钮组件 JButton 的使用方法;
- ④ 事件处理机制及动作事件的处理方法;
- ⑤ JDBC-ODBC 桥接方式连接数据库、查询数据库的一般方法。

任务一 准备数据库,定义窗体类,编制界面

【任务描述】

设计并创建一个 Access 数据库。从 JFrame 类继承出子类创建一个窗体,在其上生成界面。利用 JDBC-ODBC 桥接方式访问数据库。

【任务分析】

数据库中只需要建一张用户表。窗体上的组件可用网格布局方式组织在一个面板中。

【任务实施】

1. 准备数据库

- ① 在桌面上新建一个名称为 login 的文件夹;
- ② 在 Microsoft Access 中创建一个数据库,名称为 suserdb.mdb,保存在 login 文件夹中;
- ③ 在数据库中创建一个表,名称为 suser,表结构见表 1-1。



表 1-1 系统用户表 suser 的表结构

列名	数据类型	长度	允许为空	主键	含义
id	自动编号		否	是	id 号
username	文本	20	否		用户名
password	文本	20	否		密码

④ 在表中输入几条数据记录。

2. 编写程序框架代码

① 启动 JCreator Pro,新建一个名为 login.java 的源程序文件,保存在 login 文件夹中;

② 定义一个窗体类,类名为 login,该类从 JFrame 窗体类继承;

③ 定义构造方法,并设置窗体的大小及可见性;

④ 编写一个主方法(main()方法)。

实现代码如下:

```
//导入包
import java.awt.*;
import java.awt.event.*;
import java.sql.*;
import javax.swing.*;
//定义公共类
public class login extends JFrame
{
    //定义构造方法
    public login(String title)
    {
        super(title);

        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(500,300);
        Dimension ss = Toolkit.getDefaultToolkit().getScreenSize();
        setLocation( ( ss.width- this.getWidth() )/2,(ss.height - this.getHeight() )/2);
        setResizable(false);
        setVisible(true);
    }
    //定义主方法
    public static void main(String [] args)
    {
        new login("用户登录");
    }
}
```

3. 编制界面

(1) 定义一个面板对象及相关组件对象

将它们定义为类的成员变量,组件包括三个标签组件(JLabel)、一个文本框组件(JTextField)、两个密码框组件(JPasswordField)、两个按钮组件(JButton)。

实现代码如下:

```
JPanel p;
JLabel l1,l2,l3;
JTextField tf1;
JPasswordField tf2,tf3;
JButton bt1,bt2;
```

(2) 在构造方法中创建组件并加入至窗体中

先创建面板,将面板的布局方式设定为网格方式(GridLayout)方式,有 4 行 2 列,再创建组件,



将组件按顺序放置在面板,再将面板放置在窗体的内容窗格中。

实现代码如下:

```
p = new JPanel();
p.setLayout(new GridLayout(4,2));

l1 = new JLabel("用户名:");
l2 = new JLabel("口令:");
l3 = new JLabel("确认口令");

tf1 = new JTextField();
tf2 = new JPasswordField();
tf3 = new JPasswordField();

bt1 = new JButton("确定");
bt2 = new JButton("取消");

p.add(l1);
p.add(tf1);
p.add(l2);
p.add(tf2);
p.add(l3);
p.add(tf3);
p.add(bt1);
p.add(bt2);

getContentPane().add(p);           //可直接写成 add(p);
```

4. 加载数据库驱动程序、连接数据库

在类中定义一个用于存放数据库连接对象的静态成员变量:

```
static Connection con;
```

在主方法中加载数据库驱动程序,并连接数据库。

实现代码如下:

```
try
{
    //加载 JDBC - ODBC 驱动程序
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    con = DriverManager.getConnection("jdbc:odbc:driver = {Microsoft Access Driver (*.mdb)};DBQ = suser.mdb","","");
}
catch(ClassNotFoundException e1)
{
    JOptionPane.showMessageDialog(null,"不能识别类:" + e1);
}
catch(SQLException e2)
{
    JOptionPane.showMessageDialog(null,"连接数据库出错:" + e2);
}
catch(Exception e3)
{
    JOptionPane.showMessageDialog(null,"其他错误:" + e3);
}
```

Class 类的 `forName()` 方法在执行时可能产生的异常类型为 `ClassNotFoundException`, `DriverManager` 的 `getConnection()` 方法在执行时可能产生的异常类型为 `SQLException`, 因此在程序中对可能产生的异常进行了捕捉处理。



任务二 编制事件处理代码

【任务描述】

编制事件处理框架代码,编制“确定”、“取消”按钮代码,运行并测试程序。

【任务分析】

窗体中的两个按钮会产生动作事件,按事件处理的三个步骤分别编制代码,在“确定”按钮中要编写合法性检查及查询数据库的代码。

【任务实施】

1. 编制事件处理框架代码

① 使 login 类实现动作事件监听接口 ActionListener,主要实现代码如下:

```
public class login extends JFrame implements ActionListener
```

② 用 addActionListener() 方法对两个按钮的动作事件进行注册监听,注册监听一般在构造方法中完成。主要实现代码如下:

```
bt1.addActionListener(this);  
bt2.addActionListener(this);
```

③ 定义动作事件处理方法 actionPerformed(),编写代码框架。

```
public void actionPerformed(ActionEvent e)  
{  
    Object obj = e.getSource();  
    if(obj == bt1)  
    {  
        else if(obj == bt2)  
        {  
  
        }  
    }  
    ...  
}
```

程序中先通过 getSource() 方法获取事件源,再将“确定”、“取消”按钮的事件分开处理。

2. 编写“确定”、“取消”按钮动作事件代码

在“确定”按钮中,先取出输入的用户名、密码、两次密码,判断其合法性,即用户名、密码不能为空,并且两次输入的密码必须一致;如果不合法,则提示重新输入,合法则根据用户名及密码到用户表查找该用户是否存在,不存在则提示“数据库中不存在该用户!”的信息,存在则提示“登录成功!”的信息。实现代码如下:

```
if(tf1.getText().equals(""))  
{  
    JOptionPane.showMessageDialog(this,"用户名必须输入,不能为空!");  
    tf1.requestFocus();  
    return;  
}  
  
if((String.valueOf(tf2.getPassword())).equals(""))  
{  
    JOptionPane.showMessageDialog(this,"密码必须输入,不能为空!");
```



```
tf2.requestFocus();
return;
}
if((!(String.valueOf(tf2.getPassword())).equals(String.valueOf(tf3.getPassword()))))
{
    JOptionPane.showMessageDialog(this, "两次输入的密码不一致, 请重新输入!");
    tf3.requestFocus();
    return;
}
try
{
    Statement stat = con.createStatement();
    ResultSet result = stat.executeQuery("select * from suser where username = '" + tf1.getText() + "' and password = '" + (String.valueOf(tf2.getPassword())) + "'");

    if(result.next())
    {
        JOptionPane.showMessageDialog(this, "登录成功!");
    }
    else
    {
        JOptionPane.showMessageDialog(this, "数据库中不存在该用户!");
    }

    result.close();
    stat.close();

}
catch(SQLException e2)
{
    JOptionPane.showMessageDialog(this, "操作数据库出错:" + e2);
}
catch(Exception e3)
{
    JOptionPane.showMessageDialog(this, "其他错误:" + e3);
}
```

在“取消”按钮中,用如下代码退出程序:

```
System.exit(0);
```

3. 运行程序并进行测试

分别输入数据库表中存在和不存在的用户进行测试。

项目小结

用户登录程序是一个简单的 Java 桌面程序,通过本项目的实践,主要是为了让读者了解和掌握使用 Java 开发桌面程序的一些基本方法及步骤,掌握图形界面编制的基本方法、数据库常用的连接及查询方法、事件处理的基本步骤,为以后的开发打下基础。

项目二 简单的列表框程序

项目描述

列表框程序是一个简单的 Java 桌面应用程序,它主要用来展示不采用任何布局方式的情况下界面的编制及对列表框组件的具体操作。

项目预览

程序运行时,出现如图 2-1 所示窗体。

插入一些项及使用左移、右移后,窗体如图 2-2 所示。

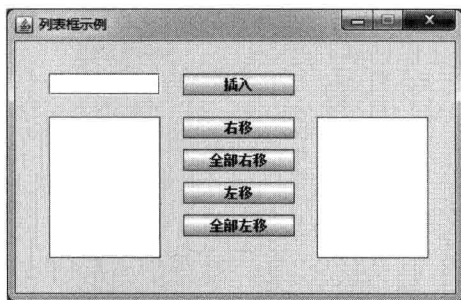


图 2-1 列表框示例窗体(1)

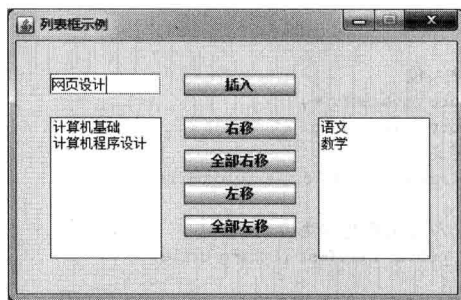


图 2-2 列表框示例窗体(2)

项目分析

可以用一个面板来组织其上的组件,列表框中项目的插入、移动主要涉及到对列表框相关方法的使用等。

环境需求

- ① 操作系统: Windows XP 或 Windows 7;
- ② JDK 版本: JDK Version 1.6;
- ③ 集成开发环境: JCreator Pro 3.5。

项目知识点

- ① Swing 中常用容器 JFrame、JPanel 的使用方法;
- ② null 布局(空布局)下组件的定位方法;
- ③ 按钮组件 JButton、列表组件 List 的使用方法;
- ④ 事件处理机制及动作事件的处理方法。

任务一 定义窗体类,编制界面

【任务描述】

定义一个窗体类,类名与主文件名同名。定义与生成相关组件来形成界面。



【任务分析】

从 JFrame 类继承出子类创建一个窗体,用一个面板来组织其上的组件面板,不采用任何布局方式,直接设置文本框、按钮及列表框组件的大小及位置生成界面。

【任务实施】

1. 编写程序框架代码

- ① 在桌面上新建一个名称为 ListTest 的文件夹;
- ② 新建一个名为 ListTest.java 的源程序文件,保存在 ListTest 文件夹中;
- ③ 定义一个公共类,类名为 ListTest,该类从 JFrame 窗体类继承;
- ④ 定义构造方法,并设置窗体的大小及可见性;
- ⑤ 编写一个主方法(main()方法)。

实现代码如下:

```
//导入包
import java.awt.*;
import javax.swing.*;
//定义公共类
public class ListTest extends JFrame
{
    //定义构造方法
    public ListTest (String title)
    {
        super(title);

        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(400,260);
        Dimension ss = Toolkit.getDefaultToolkit().getScreenSize();
        setLocation((ss.width - this.getWidth())/2, (ss.height - this.getHeight())/2);
        setResizable(false);
        setVisible(true);
    }
    //定义主方法
    public static void main(String [] args)
    {
        new ListTest ("列表框示例");
    }
}
```

2. 编制界面

- (1) 定义一个面板对象及相关组件对象为类的成员变量

组件包括一个文本框组件(JTextField)、五个按钮组件(JButton)、两个列表框组件(List)。

实现代码如下:

```
JPanel p;
JTextField tfitem;
JButton btins, bt1, bt2, bt3, bt4;
List lleft, lright;
```

- (2) 在构造方法中创建组件并加入至窗体中

先创建面板将框架的布局方式改为不采用任何布局方式,可用如下语句实现:

```
p. setLayout(null);
```

再用组件的 setBounds()方法设置组件在面板中的位置及大小,将组件先放置在面板上,再将



面板放置在窗体的内容窗格中。

主要实现代码如下：

```
p = new JPanel();
p. setLayout(null);

tfitem = new JTextField();
...
tfitem.setBounds(30,30,100,20);
...
p. add(tfitem);
...
getContentPane().add(p);           //可直接写成 add(p);
```

任务二 编制事件处理代码

【任务描述】

编制事件处理代码，运行并测试程序。

【任务分析】

窗体中的五个按钮会产生动作事件，按事件处理的三个步骤分别编制代码，“左移”、“全部左移”按钮代码与“右移”、“全部右移”按钮代码基本一致，仅数据移动方向相反。

【任务实施】

1. 编制事件处理框架代码

① 使 ListTest 类实现动作事件监听接口 ActionListener，实现代码如下：

```
public class ListTest extends JFrame implements ActionListener
```

② 用 addActionListener() 方法对五个按钮的动作事件进行注册监听，注册监听一般在构造方法中完成。实现代码如下：

```
btins.addActionListener(this);
bt1.addActionListener(this);
bt2.addActionListener(this);
bt3.addActionListener(this);
bt4.addActionListener(this);
```

③ 定义动作事件处理方法 actionPerformed()，编写代码框架。

```
public void actionPerformed(ActionEvent e)
{
    Object obj = e.getSource();
    if(obj == btins)
    {
        else if(obj == bt1)
        {
        }
    }
    ...
}
```

程序中先通过 getSource() 方法获取事件源，再将五个按钮的事件分开处理。



2. 编写五个按钮动作事件代码

在“插入”按钮中,先取出文本框中输入的文本,判断其是否为空,不为空,即可插入左边列表框,实现代码如下:

```
String sitem = tfitem.getText();
if( sitem.equals(""))
    return;
lleft.add(sitem);
```

在“右移”按钮中,先取出左边列表框中选中的项,如果有选中项,则将其插入至右边列表框中,同时删除左边列表框中的项,实现代码如下:

```
String sitem = lleft.getSelectedItem();
if( sitem != null )
{
    lright.add(sitem);
    lleft.remove(sitem);
}
```

在“全部右移”按钮中,按顺序取出左边列表框中的项,将其插入至右边列表框中,完成后删除左边列表框中的项,实现代码如下:

```
int cnt = lleft.getItemCount();
for(int i = 0; i < cnt; i++)
{
    lright.add(lleft.getItem(i));
}
for(int i = cnt - 1; i >= 0; i--)
{
    lleft.remove(i);
}
```

“左移”、“全部左移”按钮的功能与“右移”、“全部右移”类似,其代码可参照编写。

3. 运行程序并进行测试

在文本框中输入后插入列表框,进行测试。

项目小结

列表框程序也是一个简单的 Java 桌面程序,通过本项目的实践,主要是为了让读者进一步了解和掌握使用 Java 开发桌面程序的一些基本方法及步骤,掌握列表框的常见处理方法、事件处理方法等,为以后的开发打下基础。

项目三 学生成绩管理系统

项目描述

学生成绩管理系统是一个 Java 桌面数据库应用程序。它能够实现对教师、课程、班级、学生基本信息的维护,班级考试课程的选择,以及学生成绩的录入、查询、打印等功能。

项目预览

程序运行时,在用户成功登录后出现如图 3-1 所示多文档主窗体。

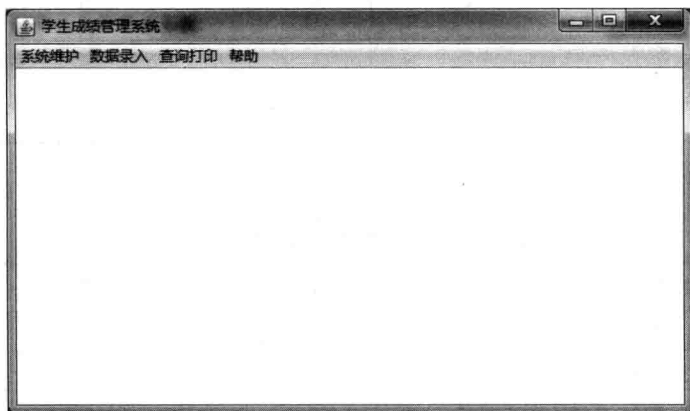


图 3-1 学生成绩管理系统主窗体

可以通过单击菜单项后打开对应的模块窗体进行操作。

项目分析

前面两个项目均是比较简单的 Java 桌面程序,主要涉及窗体、事件、数据库的查询及一些常用组件的操作。学生成绩管理系统是一个比较复杂和综合性的桌面程序,也是比较常见的数据库系统,主要功能是对一些基本实体如教师、课程、班级、学生数据的浏览及维护,班级课程的选择及学生成绩的录入、查询操作。具体开发时,将涉及到许多利用 Java 开发桌面程序的知识与技巧,包括多文档窗体、文档窗体、菜单、对话框、多种组件的编制,事件的处理,数据库的查询、插入、修改、删除操作,表格的生成及表格数据的处理等,是学习 Java 开发的理想项目。

项目设计

1. 多文档窗体的设计

应用程序界面采用 Swing 的多文档窗体界面。主窗体从 JFrame 类继承过来,上面挂有菜单,每一个菜单项打开一个文档窗体,或者称为内部框架窗体,它是从 JInternalFrame 继承下来的,它们被组织在一个被称为 JDesktopPane 类型的虚拟桌面对象上。

2. 数据库的设计

为了实现系统所要求的功能,数据库中设计了 7 张表,其中教师、课程、班级、学生、用户 5 个基本实体分别对应一张表,班级所上的课程和学生考试的成绩数据分别组织在对应的表中,有列名